问题一：

双方观测空间中不存在的状态信息是否可以作为reward？

比如两物体距离很远，双方之间无法观测到对方，全局信息中的各个智能体状态可以作为物体的reward；这里的reward该如何定义？

1. 通常思考的话，**最好只使用观测空间中的信息**来定义奖励函数。这样，智能体可以学习到一种能够充分利用观测空间中信息的策略，而不依赖于观测空间之外的信息。
2. 多智能体强化学习一种的思路为：**中心化训练，去中心化执行**。

"中心化训练，去中心化执行"策略允许在训练阶段使用观测空间之外的信息，但在执行阶段，智能体的决策应该只依赖于观测空间中的信息。

在这种策略中，智能体在**训练阶段访问全局信息**（包括其他智能体的状态和动作），但在**执行阶段只能访问局部信息**（即自己的状态和动作）。这种策略允许智能体在训练阶段学习到更有效的策略，同时在执行阶段保持了去中心化的特性。

**mpe中reward的方式**：

1. 追逐者的奖励：追逐者的奖励应该与其与逃避者的距离负相关。也就是说，当追逐者接近逃避者时，它应该得到正的奖励；反之，如果追逐者远离逃避者，它应该得到负的奖励。
2. 逃避者的奖励：逃避者的奖励应该与其与追逐者的距离正相关。也就是说，当逃避者远离追逐者时，它应该得到正的奖励；反之，如果逃避者接近追逐者，它应该得到负的奖励。
3. 碰撞的惩罚：如果追逐者与逃避者发生碰撞，那么应该给予追逐者正的奖励，但给予逃避者负的奖励。这是因为追逐者的目标是接近逃避者，而逃避者的目标是远离追逐者。
4. 其他因素：还可以考虑其他因素来调整奖励，例如追逐者的速度、逃避者的速度、追逐者和逃避者的相对方向等。

问题二：

某物体观测范围内的其他物体信息应该放在观测空间里还是reward里。

- **状态空间：**状态空间定义了智能体可以观察到的环境信息。智能体的决策是基于当前的状态进行的。状态空间中的信息应该包含智能体需要知道的所有信息，以便做出最佳的决策。例如，如果智能体需要知道其当前的位置和速度才能做出最佳决策，那么这些信息就应该包含在状态空间中。

代码上理解：**求Q值**，更新动作网络参数，能更好地进行决策。

- **奖励函数：**奖励函数定义了智能体的目标，即智能体应该如何行动才能获得最大的奖励。奖励函数通常是基于状态空间中的信息来定义的，但也可以包含其他信息。例如，如果智能体的目标是到达某个位置，那么奖励函数可能会根据智能体的当前位置和目标位置来计算奖励。

代码上理解：**TD更新**时用于目标求解。

**简略：**

**能把这个信息用到决策，放到observation里。**

**能把这个信息用到奖励求取，放到reward里。**

**题外话：**

**蓝色作为非智能体的行为模式的思考：**

想让蓝色智能体（非智能体）在没有红色智能体位置信息的情况下去追逐红色智能体，可以为蓝色智能体定义一个固定的行为模式。例如，可以让蓝色智能体按照一个**固定的路径移动(类似于巡航)**，或者**随机改变其方向(纯随机搜索)**。

固定的路径移动：

1. **直线路径：**让蓝色智能体沿着一个直线路径移动。例如，你可以让蓝色智能体从左边的边界开始，然后以固定的速度向右移动。

2. **圆形路径：**让蓝色智能体沿着一个圆形路径移动。例如，你可以让蓝色智能体以环境的中心为圆心，以固定的速度旋转。

3. **正方形路径：**以地图中心为正方形中心，让蓝色智能体绕着正方形方式运动。

其余注意的点有：

1.可能需要设定好移动速度；

2.处理边界条件（例如，当蓝色智能体到达环境的边界反弹）。

**随机改变其方向：**动作空间直接采样，这个比较好实现，直接附上具体代码：

angle\_change = np.random.uniform(-np.pi/4, np.pi/4)